

QUESTION 1: [2 Marks]

a) Discuss two main advantages of layering in computer networks? [1 Mark]

- Reduces complexity.
- Accelerates evolution.
- Simplifies teaching and learning.

b) Describe the ICMP protocol, its purpose. Where does it sit in the TCP/IP stack? [1 Mark]

b) Internet control message protocol, it is used for sending error messages indicate what is wrong. It sits in the network layer.

QUESTION 2: [3 Marks]

a) One of the main features of Java is Internet Awareness. Explain what is meant by this term. [1 Mark]

a) Means it's provide a rich, fully featured networking API that offers a consistent interface for java developers no matter what platform they are running.

b) What are exceptions? How are they handled in Java? [2 Marks]

b) Exceptions are unusual conditions that occur at runtime and are represented as objects, they are handled by three handlers:

- try: Code for throwing the exception.
- catch: to handle the exception.
- finally: to run after try or any catch.

c) Consider the following code: [2 Marks]

```
import java.net.*;
import java.io.*;

public class ip
{
    public static void main ( String[] args ) throws IOException
    {
        String hostname = args[0];

        try
        {
            InetAddress ipaddress = InetAddress.getByName(hostname);
            System.out.println("IP address: " + ipaddress.getHostAddress());
        }
        catch ( UnknownHostException e )
        {
            System.out.println("Could not find IP address for: " + hostname);
        }
    }
}
```

i. Using the command-line, compile and run the code?

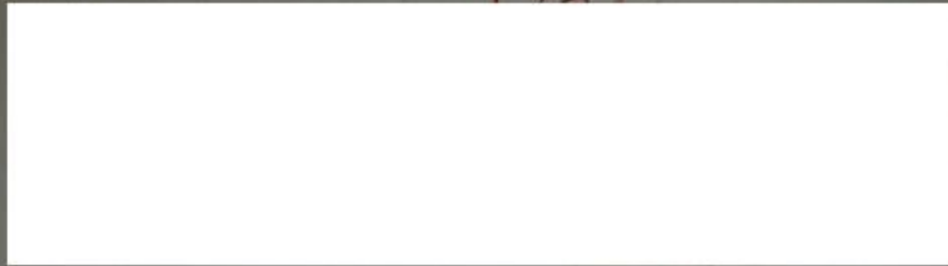
ii. What will be the output?

QUESTION 3: [2 Marks]

a) Briefly describe what is meant by Object Persistence? [1/2 Mark]

b) How to make a Java class Serializable? [1/2 Mark]

- c) While serializing you want some of the members not to serialize, why? How do you achieve it? [1 Mark]



QUESTION 4: [7 Marks]

- a) UDP doesn't provide delivery guarantee at protocol level. How do you implement reliable transmission in UDP protocol? [1 Mark]



- b) Briefly describe what is meant by a socket, and what class is used to create a socket in Java? [1 Mark]

b) A mechanism that provides an endpoint for communication, "DatagramSocket".

- c) The client in socket programming must know two things about the server, what are they? [1 Mark]

c) The IP address and the port number.

- d) There are two reasons to create DatagramPacket, what are they? [1 Mark]

d). To send data to a remote machine using UDP.
• To receive data sent by a remote machine using UDP.

- e) The following is an example of sending DatagramPacket by DatagramSocket, comment the specified lines (marked with "//")? [2 Marks]

```
//DSender.java
```

```
//
```

```
import java.net.*;
```

```
public class DSender(  
    public static void main(String[] args) throws Exception {
```

```
//
```

```
DatagramSocket ds = new DatagramSocket();  
String str = "Welcome java";
```

```
//
```

```
InetAddress ip = InetAddress.getByAddress("127.0.0.1");
```

```
//
```

```
DatagramPacket dp = new DatagramPacket(str.getBytes(), str.length(), ip, 3000);
```

```
//
```

```
ds.send(dp);
```

```
//
```

```
ds.close();
```

```
}
```

- f) The following is an example of receiving DatagramPacket by DatagramSocket, comment the specified lines (marked with "//")? [1 Marks]

```
//DReceiver.java  
  
import java.net.*;  
  
public class DReceiver{  
    public static void main(String[] args) throws Exception {  
  
        //  
        DatagramSocket ds = new DatagramSocket(3000);  
  
        //  
        byte[] buf = new byte[1024];  
  
        //  
        DatagramPacket dp = new DatagramPacket(buf, 1024);  
  
        ds.receive(dp);  
        String str = new String(dp.getData(), 0, dp.getLength());  
        System.out.println(str);  
        ds.close();  
    }  
}
```


QUESTION 2: [4 Marks]

a) Find the netid and the hostid of the following IP addresses. [1 Mark]

a. 81.44.12.6

b. 212.31.55.08

b) How can I find out the current IP address for my machine? [1 Mark]

//write the code

1.
2.

b) `InetAddress a = InetAddress.getLocalHost();`
`System.out.print("My IP address is: " + a.getHostAddress());`

2. Answer with **True** or **False** in the following: [4 Marks]

- a. (~~F~~) A socket may be connected to more than one host at a time and socket may not reconnect after it's closed.
- b. (~~F~~) The `java.net.Socket` class allows you to create socket objects that perform all four fundamental socket operations: Connect to a remote machine, Send data, Receive data and Close the connection.
- c. (~~T~~) The `Socket()` constructors do not just create a `Socket` object. They also attempt to connect the underlying socket to the remote server.
- d. (~~F~~) A server socket binds to a randomly selected port on the local machine. Once it has successfully bound to a port, it listens for incoming connection attempts.
- e. (~~F~~) When a server detects a connection attempt, it *accepts* the connection. This creates a socket between the client and the server over which the client and the server communicate.
- f. (~~F~~) Multiple clients can connect to the same port on the server at the same time.

- g. (F) Constructing a server socket is as simple as specifying the port you want to listen on, like this:

```
try {  
    ServerSocket ss = new ServerSocket(80);  
}  
catch (IOException e) {  
    System.err.println(e);  
}
```

- h. (F) When a `ServerSocket` object is created, it attempts to *bind* to the port on the local host given by the port argument. If another server socket is already listening to the port, then it waits until the connection closed and tries to bind again.

QUESTION 3: [5 Marks]

- a) How do I get the IP address of a machine (remote) from its hostname? [1 Mark]

//write the code

- 1.
- 2.

- b) How do I perform a hostname lookup for an IP address? [1 Mark]

//write the code

- 1.
- 2.

- c) How can I find out the current IP address for my machine? [1 Mark]

//write the code

1. `InetAddress ad = InetAddress.getLocalHost();`
2. `System.out.println("IP Ad = " + ad.getHostAddress());`

- d) Consider the following code: [2 Marks]

```
1 //InetDemo.java  
2 import java.io.*;  
3 import java.net.*;  
4  
5 public class InetDemo{  
6     public static void main(String[] args){  
7         try{  
8             InetAddress ip=InetAddress.getByName("www.javatpoint.com");  
9             System.out.println("Host Name: " + ip.getHostName());  
10            System.out.println("IP Address: " + ip.getHostAddress());  
11        }  
12        catch(Exception e){  
13            System.out.println(e);  
14        }  
15    }  
16 }
```

1. Using the command-line, compile and run the code?

2. What will be the output if the correspondent IP address of `www.javatpoint.com` is `206.51.231.148`?

Host Name: `www.javatpoint.com`

IP Address: `206.51.231.148`

c) Briefly discuss two commonly used constructors by DatagramSocket class, and two commonly used constructors by DatagramPacket class? [2 Marks]

d) The following is an example of receiving DatagramPacket by DatagramSocket, comment the specified lines (marked with "//")? [1 Mark]

```
//DReceiver.java name of the file importing net package  
import java.net.*;
```

```
public class DReceiver{  
    public static void main(String[] args) throws Exception {
```

```
        // Creating a socket with port number 3000.
```

```
        DatagramSocket ds = new DatagramSocket(3000);
```

```
        // creating a data buffer of byte array with size 1024
```

```
        byte[] buf = new byte[1024];
```

```
        // creating a datagram packet with byte array.
```

```
        DatagramPacket dp = new DatagramPacket(buf, 1024);
```

```
        // receiving packets using dp as a container.
```

```
        ds.receive(dp);
```

```
        String str = new String(dp.getData(), 0, dp.getLength());
```

```
        System.out.println(str);
```

```
        ds.close();
```

```
    }
```

```
}
```

Q3. Consider the queuing delay in a router buffer (preceding an outbound link). Suppose all packets are L bits, the transmission rate is R bps, and that N packets simultaneously arrive at the buffer every LN/R seconds. Find the average queuing delay of a packet. (Hint: The queuing delay for the first packet is 0; for the second packet L/R ; for the third packet $2L/R$. The N th packet has already been transmitted when the second batch of packets arrive)

[3 Marks]

QUESTION 1:

Answer with **True** or **False** in the following: **[4 Marks]**

- a. (~~T~~) When a server detects a connection attempt, it accepts the connection. This creates a socket between the client and the server over which the client and the server communicate.
- b. (~~T~~) Multiple clients can connect to the same port on the server at the same time.
- c. (~~T~~) Constructing a server socket is as simple as specifying the port you want to listen on, like this:

```
try {  
    ServerSocket ss = new ServerSocket(80);  
}  
catch (IOException e) {  
    System.err.println(e);  
}
```

- d. (~~F~~) When a ServerSocket object is created, it attempts to bind to the port on the local host given by the port argument. If another server socket is already listening to the port, then it waits until the connection closed and tries to bind again.
- e. (~~F~~) A socket may be connected to more than one host at a time and socket may not reconnect after it's closed.
- f. (~~T~~) The java.net.Socket class allows you to create socket objects that perform all four fundamental socket operations: Connect to a remote machine, Send data, Receive data and Close the connection.
- g. (~~T~~) The Socket() constructors do not just create a Socket object. They also attempt to connect the underlying socket to the remote server.
- h. (~~F~~) A server socket binds to a randomly selected port on the local machine. Once it has successfully bound to a port, it listens for incoming connection attempts.